

(12) Japanese Patent Publication

(11) 2003-536153

(43) 2003.12.2.

(21) 2002-502720

(86)(22) 2001.6.8

(71) IMAGINATION TECHNOLOGIES LIMITED [GB/GB]; Home Park Estate Kings Langley Hertfordshire WD4 8L2 (GB).

MORPHET, Stephen [GB/GB]; 80a Weymouth Street Hemel Hempstead Herts HP3 9SH (GB).

(72) MORPHET, Stephen [GB/GB]; 80a Weymouth Street Hemel Hempstead Herts HP3 9SH (GB).

(54) TILING AND COMPRESSION FOR RENDERING 3D IMAGES

(57) A 3-dimensional computer generated image is generated by subdividing the image into a plurality of rectangular areas. Object data for each rectangular area is loaded into a display list memory (4) until that memory is substantially full. Image data and shading data for each picture element of each rectangular area is derived by an image synthesis processor (6) from the object data. Image data is then stored in a local memory (16) and further object data loaded into the display list memory (4) and replaces the existing contents. Once this has happened, the stored image data and shading data is retrieved and additional image data and shading data derived for each picture element by the image synthesis processor (6) using the new object data and the previously derived image and shading data. When there is no further object data to load to the display list memory the shading data is provided for display for the rectangular areas by a frame buffer (11). The memory may be managed by allocating at least one block of storage from the display list memory to each rectangular area and then storing in that block of memory, data which pertains to surfaces which intersect that rectangular area. A determination is made as to when a predetermined number of blocks have been used for a rectangular area. When this number is reached, the system starts to derive shading data for the rectangular areas, thereby releasing blocks of storage to be allocated to further rectangular areas in the display list memory.

BEST AVAILABLE COPY

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号
特表2003-536153
(P2003-536153A)

(43) 公表日 平成15年12月2日 (2003. 12. 2)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 T 15/00	1 0 0	G 0 6 T 15/00	1 0 0 A 5 B 0 8 0

審査請求 未請求 予備審査請求 未請求 (全 44 頁)

(21) 出願番号 特願2002-502720 (P2002-502720)
(86) (22) 出願日 平成13年6月8日 (2001. 6. 8)
(85) 翻訳文提出日 平成14年2月8日 (2002. 2. 8)
(86) 国際出願番号 P C T / G B 0 1 / 0 2 5 3 6
(87) 国際公開番号 W O 0 1 / 0 9 5 2 5 7
(87) 国際公開日 平成13年12月13日 (2001. 12. 13)
(31) 優先権主張番号 0 0 1 4 3 0 7 . 6
(32) 優先日 平成12年6月8日 (2000. 6. 8)
(33) 優先権主張国 イギリス (G B)
(31) 優先権主張番号 0 0 2 7 8 9 7 . 8
(32) 優先日 平成12年11月15日 (2000. 11. 15)
(33) 優先権主張国 イギリス (G B)

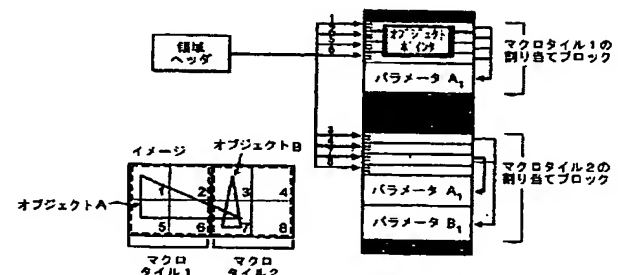
(71) 出願人 イマジネーション テクノロジーズ リミ
テッド
イギリス ハートフォードシャー ダブリ
ューディー4 8エルゼット キングス
ラングリー ホーム パーク エステイト
(番地なし)
(71) 出願人 モーフエット スティーブン
イギリス ハーツ エイチビー3 9エス
エイチ ヘメル ヘンプステッド ウエイ
マス ストリート 80エイ
(74) 代理人 弁理士 中村 稔 (外9名)

最終頁に続く

(54) 【発明の名称】 三次元イメージレンダリングのためのタイリング及び圧縮

(57) 【要約】

三次元コンピュータ生成イメージは、イメージを複数の矩形領域に細分化することにより生成される。各々の矩形領域のオブジェクトデータは、ディスプレイリストメモリ (4) が実質的に満杯になるまで、そのメモリにロードされる。各々の矩形領域の各々の画素のイメージデータ及びシェイディングデータは、イメージ合成プロセッサ (6) によってオブジェクトデータから引き出される。次に、イメージデータは、ローカルメモリ (16) に記憶され、別のオブジェクトデータは、ディスプレイリストメモリ (4) にロードされ、既存の内容と置き換えられる。これが起こると、記憶されているイメージデータ及びシェイディングデータが検索され、新しいオブジェクトデータと以前に引き出されたイメージデータ及びシェイディングデータとを使用して、イメージ合成プロセッサ (6) によって、各々の画素に関する追加のイメージデータ及びシェイディングデータが引き出される。ディスプレイリストメモリにロードする別のオブジェクトデータがない場合は、フレームバッファ (11) によって矩形領域に関するシェイディングデータが表示



【特許請求の範囲】

【請求項1】 三次元コンピュータイメージを生成する方法であって、

- a) イメージを複数の矩形領域に細分化する段階と、
 - b) ディスプレイリストメモリが実質的に満杯になるまで、各々の矩形領域のオブジェクトデータを前記ディスプレイリストメモリにロードする段階と、
 - c) 前記オブジェクトデータから各々の矩形領域の画素のイメージデータ及びシェイディングデータを引き出す段階と、
 - d) 前記イメージデータ及びシェイディングデータを記憶する段階と、
 - e) 別のオブジェクトデータを前記ディスプレイリストメモリにロードして既存の内容と置き換える段階と、
 - f) 記憶されているイメージデータ及びシェイディングデータを検索する段階と、
 - g) 前記新しいオブジェクトデータと前記先に引き出されたイメージデータ及びシェイディングデータとから、前記各々の矩形領域の各々の画素に関する追加のイメージデータ及びシェイディングデータを引き出す段階と、
 - h) 前記ディスプレイリストメモリにロードするオブジェクトデータがなくなるまで、段階d)、段階e)、段階f)、及び段階g)を繰り返す段階と、
 - i) 表示するために前記シェイディングデータを供給する段階と、
- を含む方法。

【請求項2】 前記イメージデータが、オブジェクト識別データと奥行きデータとを含むことを特徴とする請求項1に記載の方法。

【請求項3】 段階d)の前に前記イメージデータを圧縮する段階と、段階g)の前に前記圧縮されたイメージデータを圧縮解除する段階とを含むことを特徴とする請求項1又は2に記載の方法。

【請求項4】 三次元コンピュータイメージを生成する装置であって、

- a) イメージを複数の矩形領域に細分化する手段と、
- b) ディスプレイリストメモリが実質的に満杯になるまで、各々の矩形領域のオブジェクトデータを前記ディスプレイリストメモリにロードする手段と、
- c) 前記オブジェクトデータから各々の矩形領域の画素のイメージデータ及びシ

エイディングデータを引き出す手段と、

d) 各々の矩形領域の前記イメージデータ及びシェイディングデータを記憶する手段と、

e) 別のオブジェクトデータを前記ディスプレイリストメモリにロードして既存の内容と置き換える手段と、

f) 記憶されているイメージデータ及びシェイディングデータを検索する手段と、

g) 前記新しいオブジェクトデータと、前記記憶されているイメージデータ及びシェイディングデータとから、各々の矩形領域の各々の画素に関する別のイメージデータ及びシェイディングデータを引き出す手段と、

h) 前記ディスプレイリストメモリにロードするオブジェクトデータがなくなるまで、構成要件d)、e)、f)、及びg)を繰り返して機能させる手段と、

i) 表示のために前記シェイディングデータを供給する手段と、
を備えることを特徴とする装置。

【請求項5】 前記イメージデータが、オブジェクト識別データ及び奥行きデータを備えることを特徴とする請求項4に記載の装置。

【請求項6】 記憶される前に前記イメージデータを圧縮する手段と、検索された後に前記圧縮されたイメージデータを圧縮解除する手段とを備えることを特徴とする請求項4又は5に記載の装置。

【請求項7】 イメージデータを複数の実質的に矩形領域に細分化する手段と、前記イメージを作成する面に関連するデータをディスプレイリストメモリに記憶する手段と、少なくとも1つの記憶ブロックを前記ディスプレイリストメモリから各々の矩形領域に割り当てる手段と、前記矩形領域と交差する面に関連するデータを前記メモリブロックに記憶する手段と、前記各々の矩形領域のデータを前記ディスプレイリストから前記矩形領域の各々の画素のシェイディングデータを引き出す手段へ供給する手段と、表示のために前記シェイディングデータを記憶するフレーム記憶手段と、を備える三次元コンピュータイメージを生成するシステムと共に使用されるメモリ管理システムであって、前記各々の矩形領域が、複数の小さな矩形領域を含み、前記シェイディングデータを引き出す手段が、

前記各々の小さな矩形領域の前記データを順番に引き出すことを特徴とするメモリ管理システム。

【請求項8】 前記記憶ブロックをディスプレイリストメモリから割り当てる手段は、所定数のブロックが使用されている場合を判定し、前記判定結果に基づいて、前記データを供給する手段に、前記シェイディングデータを引き出す手段が操作を開始するようにさせて、これにより、別のオブジェクトデータのために記憶ブロックを解放するようになっていることを特徴とする請求項7に記載のメモリ管理システム。

【請求項9】 前記イメージデータが、一連のデータフレームを備えることを特徴とする請求項7又は8に記載のメモリ管理システム。

【請求項10】 イメージを複数の矩形領域に細分化する段階と、前記各々の矩形領域と交差する前記イメージの面に関連するデータをディスプレイリストメモリに記憶する段階と、前記ディスプレイリストメモリの少なくとも1つのブロックを前記各々の矩形領域に割り当てる段階と、前記各々の矩形領域と交差する面に関連するデータを前記ブロックに記憶する段階と、各々の矩形領域のデータをシェイディング手段に供給して、シェイディングデータが前記矩形領域を作成する各々の画素から引き出されるようにする段階と、表示のために前記シェイディングデータを供給する段階を含む、三次元コンピュータイメージを生成するシステムのメモリ管理方法であって、前記イメージを細分化する段階は、前記各々の矩形領域を複数の小さな矩形領域に細分化する段階を更に含み、前記シェイディング手段は、順番に前記各々の小さな矩形領域をシェイディングすることを特徴とする方法。

【請求項11】 前記ディスプレイリストメモリの所定数のブロックが前記矩形領域に割り当てられている場合を判定し、前記判定結果に基づいて、前記シェイディング手段へのデータの供給を開始して、これにより、別のオブジェクトデータのために記憶ブロックを解放する段階を含む請求項10に記載のメモリ管理方法。

【請求項12】 イメージデータを複数の実質的に矩形領域に細分化する手段と、前記イメージを作成する面に関連するデータをディスプレイリストメモリ

に記憶する手段と、少なくとも1つの記憶ブロックを前記ディスプレイリストメモリから各々の矩形領域に割り当てる手段と、前記矩形領域と交差する面に関連するデータを前記メモリブロックに記憶する手段と、各々の矩形領域のデータを前記ディスプレイリストから前記矩形領域の各々の画素のシェイディングデータを引き出す手段へ供給する手段と、表示のために前記シェイディングデータを記憶するフレーム記憶手段と、を備える三次元コンピュータイメージを生成するシステムと共に使用されるメモリ管理システムであって、前記記憶ブロックをディスプレイリストメモリから割り当てる手段は、所定数のブロックが使用されている場合を判定し、前記判定結果に基づいて、前記データを供給する手段に、前記シェイディングデータを引き出す手段が操作を開始するようにさせて、これにより、別のオブジェクトデータのために記憶ブロックを解放するようになっていることを特徴とするメモリ管理システム。

【請求項13】 イメージを複数の矩形領域に細分化する段階と、前記各々の矩形領域と交差する前記イメージの面に関連するデータをディスプレイリストメモリに記憶する段階と、前記ディスプレイリストメモリの少なくとも1つのブロックを前記各々の矩形領域に割り当てる段階と、前記各々の矩形領域と交差する面に関連するデータを前記ブロックに記憶する段階と、各々の矩形領域のデータをシェイディング手段へ供給して、シェイディングデータが前記矩形領域を作成する各々の画素から引き出されるようにする段階と、表示のために前記シェイディングデータを供給する段階を含む、三次元コンピュータイメージを生成するシステムのメモリ管理方法であって、前記ディスプレイリストメモリの所定数のブロックが前記矩形領域に割り当てられている場合を判定し、前記判定結果に基づいて、前記シェイディング手段へのデータの供給を開始して、これにより、別のオブジェクトデータのために記憶ブロックを解放する段階を含むことを特徴とする方法。

【請求項14】 実質的に本明細書で説明するような三次元コンピュータ生成イメージをシェイディングするシステムと共に使用されるメモリ管理システム。

【請求項15】 実質的に本明細書で説明するような三次元コンピュータ生

(6)

特表 2003-536153

成イメージをシェイディングするシステムのメモリ管理方法。

【発明の詳細な説明】**【0001】**

本発明は、三次元コンピュータ生成イメージを生成するシステムと共に使用するメモリ管理装置に関する。

【0002】

本出願人による英国特許第2281682号には、ポリゴン三次元イメージ作成システムが説明されており、各々のオブジェクトは、一連の無限の面によって表示される。イメージが表示される画面の各々の基本領域は、これを通して視点から三次元シーンへ投影される光線を受ける。投射された光線が各々の面と交差する位置が決定される。これらの交差位置から、任意の交差面がその基本領域において見えるか否かを決定することが可能である。その決定結果に基づいて基本領域が表示のためにシェイディングされる。

【0003】

このシステムは、複数のセルを備えるパイプライン型プロセッサで実行でき、各々が面との交差位置の計算を行うことができる。つまり、多数の面交差位置を同時に計算できる。各々のセルは、交差位置テストを行う面を定義する一連の係数をロードする。

【0004】

英国特許出願第2298111号には改良されたシステムが説明されている。この出願において、イメージ平面は、タイルを順番に処理できるようにサブ領域又はタイルに細分化されている。可変タイルサイズを使用して、複雑なオブジェクトの回りに境界ボックスを投影することが提案されており、境界ボックスの範囲に含まれるタイルのみを処理するようになっている。このことは、最初に、適切なタイルサイズを選択するために、可視画面上のオブジェクト分布を決定することにより行われる。次に、種々のオブジェクトを定める面が、ディスプレイリストとして知られているリストに記憶され、これにより、多数の面で作られた1つのオブジェクトが複数のタイルに現れる場合があるので、各々のタイルに対して同一の面を記憶することが回避される。ディスプレイリストのオブジェクトを特定するオブジェクトポインタも同様に記憶されている。タイル毎に1つのオブ

ジェクトポインタリストが存在する。また、前述の投光手法を利用して各々のタイル内の全てのオブジェクトが処理されるまで、タイルは順番にレンダリングされる。このことは、特定のタイルでは見えないことが分かっているオブジェクトをレンダリングする必要がないので好都合である。

【0005】

国際特許出願第PCT/GB99/03707には他の改良されたシステムが提案されており、特定のオブジェクトを表示する必要がない境界ボックスのタイルは、レンダリングの前に破棄される。

【0006】

図1には、使用するプロセッサ形式のブロック図が示されている。これはタイルアクセラレータユニット2を備えており、前述のタイリング操作を行ってタイル情報をディスプレイリストメモリ4へ供給するようになっている。次に、処理のためのタイルオブジェクトデータは、イメージ合成プロセッサ（ISP）6に供給され、前述の光線／面交差位置のテストを行うようになっている。これはオブジェクト識別データ及び奥行きデータを含むイメージデータをもたらす。その後、このイメージデータは、テクスチャリング／シェイディングプロセッサ（TSP）8に供給され、テクスチャリング及びシェイディングデータを目に見えると決定されている面に適用し、イメージシェイディングデータをフレーム記憶装置へ出力するようになっている。

【0007】

多くの三次元グラフィックシステムは、即時モードレンダリングとして知られている手法を利用する。この形式のレンダリングは、ディスプレイリストメモリ内のシーン全体のコピーを記憶する必要がない。その代わりに、各々のポリゴンは、レンダリングハードウェアに送られると即座にレンダリングされる。このことは、ディスプレイリストメモリの記憶要件を回避するという利点がある。しかし、本出願人の英国特許第2281682号に説明され、英国特許出願第2298111号で改良された形式のレンダリング手法で可能な注目に値する最適化を不可能にする。据え置きテクスチャ法及び透過ソート法などの利点は、この種のアーキテクチャを使用すれば可能である。

【0008】

三次元グラフィックシステムを使用してレンダリングする必要があるシーンの複雑さ、ハードウェアが強力になるに従って増してくる。しかし、このことは、ディスプレイリストの記憶要件も高くなるので、ディスプレイリストを基本にしたレンダリングシステムに関する問題点を提示する。従来、ディスプレイリストメモリの空きがなくなれば、単にシーンの一部がレンダリングされないだけであった。この種のイメージ破壊はもはや容認できないと考えられている。シーンの複雑さが増すのに追従してメモリのコストは安くなっていないので、システムのメモリ量を増やすにも限界がある。

【0009】

本発明の好適な実施形態は、メモリ消費量に上限を設けるが、同時にシステムにより消費されるメモリ帯域幅を最小限に抑えることを試みる方法で、ディスプレイリストを管理する方法を提供する。このことは、タイルレンダリングが終了する前にシステム（ISP及びTSP）の状態をメモリに記憶して、レンダリングを終了するために後でこの状態を再ロードすることにより達成できる。この手法を「z／フレームバッファのロード／記憶」と呼ぶ。

【0010】

好適な実施形態において、この画面は、マクロタイルと呼ぶ複数の領域に分割され、各々は複数の小さなタイルで作られた、画面の矩形領域から構成されている。次に、ディスプレイリスト内のメモリは、ブロックに分割されて、空き記憶域リストに記載される。次に、空き記憶域リストのブロックは、必要に応じてマクロタイルに割り当てられる。タイリング操作により、ポリゴンパラメータデータと、目に見える各々のマクロタイルに関連する各々のブロックについての面オブジェクトポインタが記憶される。パラメータのためのメモリが満杯になるか、又は所定の閾値に達すると、システムは、マクロタイルを選択し、z／フレームバッファロードを実行して、マクロタイルの内容をレンダリングした後に、これをz／フレームバッファ記憶操作を利用して記憶する。レンダリングが終了すると、システムは、マクロタイルに関連する任意のメモリブロックを解放するので、それらのメモリブロックは、別のパラメータの記憶に利用可能になる。z／フ

レームバッファのロード及び記憶操作は、従来のように画面上の全てのタイルではなく、実際にレンダリングされるマクロタイルに制限される。次に、表示される現在のフレーム又は次のフレームのいずれかのタイル処理は、マクロタイルのレンダリング及び同一プールから別のマクロタイルへのメモリブロックの割り当てに並行して続けられる。パラメータリストの二重バッファリングを行う必要がないので、システムのメモリ要件が更に軽減される。

【0011】

本発明は、参照すべき添付の特許請求の範囲に更に明確に定義されている。

以下に本発明の好適な実施形態を、添付図面を参照して例示的に詳細に説明する。

【0012】

図1のシステムにおいて、表示されるポリゴンは、平面の三角形に分解される。これらを表すデータは、ハードウェア又はソフトウェアのいずれかによって画面空間表現に変換されて、ローカルメモリに記憶される。次に、タイルアクセラレータ2により実行されるタイリングプロセスにより、画面内の各々のタイルの変換三角形データに関するポインタリストが作成される。次に、このリストは、三角形が一部となっているオブジェクトに関連するパラメータと共に、ディスプレイリストに記憶される。このことは図2に概略的に示されており、領域ヘッダ10は、図示のタイル1から8の識別に対応する。図示のように、タイルは、オブジェクトA及びオブジェクトBという2つの三角形を覆っている。オブジェクトAは、タイル1、2、5、6、7に見ることができ、オブジェクトBは、タイル3、7に見ることができる。

【0013】

ディスプレイリスト12は、タイルから各々のタイル内に見ることができるオブジェクトデータを受け取り、各々のオブジェクトに関連するパラメータ、即ち、この例ではパラメータAとパラメータBとのリンク付けが行なわれる。つまり、タイル1、2、5、6、7のオブジェクトポインタはパラメータAを指し、タイル3、7のオブジェクトポインタはパラメータBを指すことが分かる。このデータがイメージ合成プロセッサ6及びテクスチャリング/シェイディングプロセ

ッサ8に供給されると、正しいイメージが各々のタイル画素に適用される。

【0014】

ディスプレイリストの大きさに対してイメージが複雑な場合、ディスプレイリストの全タイルデータを同時に記憶できない場合があるので、z／フレームバッファのロード／記憶と呼ぶ手法を使用する。その結果、ディスプレイリストには、データが、実質的に満杯になるまでタイルアクセラレータによってロードされる。これはイメージデータの一部を示すに過ぎない。次に、このデータは、ISP及びTSPによってレンダリングされ、その結果はローカルメモリに記憶される。これを部分レンダリングと呼ぶ。イメージは、通常の方法で一度に1つのタイルがレンダリングされる。各々のタイルのレンダリング後に、ISPの内部状態（z、ステンシル、及びオブジェクトアドレス情報を含む）が外部メモリに書き込まれ、TSPからのイメージデータがフレームバッファに書き込まれる。これら2つの操作は一緒になってz／フレームバッファの記憶操作を構成する。部分レンダリングの終わりに、画面上の各々のタイルに対するレンダリング及びz／フレームバッファ記憶操作が発生している。

【0015】

この時点で、ディスプレイリストの第1の部分が破棄され（このデータはタイルをレンダリングするのに使用した）、メモリは、タイルアクセラレータから別の出力を受け取るために使用される。ディスプレイリストメモリが再度満杯になるか、又はタイルアクセラレータがシーンデータの終わりに達すると、別の部分レンダリングが発生する。各々のタイルに対して順番に処理が始まると、先に記憶したISPデータ情報は外部メモリから再ロードされ、必要であれば、混合操作を行うために、フレームバッファからのイメージデータをTSPにロードする。これがz／フレームバッファロード操作である。従って、あたかも割り込みを受けていないかのように、そのタイル内でディスプレイリストの次の部分をレンダリングできる。最後に、タイルがレンダリングされた後に、z／フレームバッファ記憶は、必要なデータ情報を記憶するので、別のサイクルの部分レンダリングを発生可能になる。

【0016】

このように作動することにより、限られたメモリ量だけを使用して非常に複雑なシーンのレンダリングが可能になる。

【0017】

図3は、本発明の実施形態のディスプレイリストのメモリがどのように割り当てられるかを示す。図3において、図1に示すイメージ部は、この例ではマクロタイル1及びマクロタイル2という見出しを付けた2つのマクロタイルに分割されていることが分かる。ディスプレイリストは、2つの部分に分割されており、第1の部分はマクロタイル1に関し、第2の部分はマクロタイル2に関するものである。マクロタイル1のタイルの領域ヘッダは、全てマクロタイル1のオブジェクトポインタに入る。マクロタイル1ではオブジェクトAしか見えないので、パラメータA₁のみをこのメモリブロックに記憶すればよい。従って、タイル1、2、5、6のオブジェクトポインタは、全てパラメータA₁を指していることが分かる。

【0018】

マクロタイル2ではオブジェクトA及びオブジェクトBの両方が見えており、両方ともタイル7に存在している。従って、パラメータA₂及びパラメータB₂は、マクロタイル2の割り当てメモリブロックに記憶される必要がある。結果的に、タイル7のオブジェクトポインタは、パラメータA₂及びパラメータB₂を指し、一方で、タイル3のオブジェクトポインタは、パラメータB₂のみを指していることが分かる。

【0019】

マクロタイルのサイズは予め決定でき、又はシーンの複雑性に応じてマクロタイルのサイズを定義するシステムを実行できる。マクロタイリングプロセスは、タイルアクセラレータ2による基本タイリング操作の下流側で実行できる。

【0020】

ディスプレイリストの記憶に利用可能なメモリは、最初に割り当てブロックに分割され、各々のブロックのアドレスは、FIFO又は空き記憶域として知られるスタックに記憶される。ブロックサイズの選択は、大きなブロックによる空間損失量と、多数の小さなブロックの管理に関連する記憶要件との間の妥協により

なされる。また、利用するメモリ技術の特性にも影響される。

【0021】

既にタイルに分割されている画面は、次に、マクロタイルに分割される。最も単純な形態において、画面上の全てのマクロタイルは、同じ寸法であり、最小のものが1つのタイルに相当する。図3に示す例では、説明を簡単にするためにマクロタイルサイズ2 x 2を用いているが、実際には、そのサイズは大きくてもよい。

【0022】

最初に、全ての割り当てブロックは、空き記憶域（図示せず）にある。タイルアクセラレータは、三角形等の特定のオブジェクトが交差する1組のタイル（及びマクロタイル）を識別する。次に、マクロタイルと交差する第1のオブジェクトデータが識別されると、各々のマクロタイルについて、順番に、空き記憶域からのブロックが要求され、そのマクロタイルと関連づけられる。次に、オブジェクトデータは、そのブロックに記憶される。現行のブロックで1つのデータを記憶する空間が不足すると、空き記憶域からの更なるブロックが要求される。

【0023】

イメージ合成プロセッサ6及びテクスチャリング／シェイディングプロセッサ8によってマクロタイルがレンダリングされると、関連のメモリブロックを空き記憶域に戻すことができる。そのためには、ブロックを何とかして関連マクロタイルとリンク付けする必要がある。これを行う1つの可能な方法は、各々の割り当てブロックにおいてワードを利用してブロックのリンクリストを生成することである。もしくは、空き記憶域リストからの割り当てブロックを追加及び取り除くのではなく、空き記憶域構造内で生成される複数の使用ブロックの間のリンクを用いて、現場で使用中又は未使用とマーキングすることができる。

【0024】

前述のように、タイルアクセラレータは、通常の方法で1組のタイル及び三角形のストリップのストリップマスク情報を生成するのに使用される。更に、タイルアクセラレータは、ストリップがどのマクロタイルに該当するか決定する。次に、タイルアクセラレータ2は、ストリップが見える各々のマクロタイルの独立

したリストにパラメータデータを書き込む。このパラメータデータは、各々のマクロタイルに関してディスプレイリスト内の適切な位置に書き込まれる。データの何らかの重複が起こることは明らかである。しかし、このことは、他のマクロタイルのディスプレイリストに影響を与えることなく、マクロタイルがレンダリングされた後に、特定のマクロタイルに関連する全てのメモリを解放することを可能にする。

【0025】

表示されるイメージデータの第1のフレームの最初の部分でタイリングが始まる。各々のマクロタイルにおいて見えるオブジェクトの、オブジェクトポインタが生成され、マクロタイルにおいて見える各々のオブジェクトの、該当オブジェクトパラメータと共に、マクロタイルに割り当てられたディスプレイリストのブロックに書き込まれる。このプロセスは、ディスプレイリストが実質的に満杯になるか、又は所定の閾値に達するまで、例えば、割り当てブロックの75%が使用されるまで継続する。トリガーレベルに達すると、システムは、イメージ合成プロセッサ及びテクスチャリング／シェイディングプロセッサを使用して、マクロタイルを選択し、そのレンダリングを開始してフレームバッファへ出力する。マクロタイルの選択方法を以下に詳細に説明する。その判定は、複数の要因に基づいていてもよい。典型的に、新しいデータのための最大量のメモリを解放するので、その時点で最大数の割り当てブロックを使用しているマクロタイルが選択されることになる。タイリングは、第1のフレームのデータの残りのレンダリングと並行して継続する。75%閾値が使用される。その理由は、これにより、継続中のタイリングプロセスによって割り当てられる十分な未使用メモリが残り、一方で、他のメモリがレンダリングされて空き記憶域に戻されるからである。レンダリング中のマクロタイルへタイリングを行い得るように、その領域ヘッダ（マクロタイルのポインタリスト及び割り当てブロックに対するポインタ）は、空のタイルに対するポインタと交換される。従って、以前のマクロタイルのデータに対するレンダリングを継続でき、タイルアクセラレータは、空のタイルのための新しい割り当てブロックにデータを書き込むことができる。マクロタイルのレンダリングが終了すると、割り当てブロックは、空き記憶域に戻される。従って

、メモリ使用量が閾値を超えている場合は別のマクロタイルが選択され、別のレンダリング操作が始まる。75%閾値は調整可能なパラメータである。その理由は、ISP及びTSPは、ディスプレイリストの大部分がメモリ内にある場合に効率的に（リスト全体がメモリ内にあると最も効率的に）据え置きテクスチャリング等の操作を行うことができるからである。閾値は、大きなディスプレイリストサイズの矛盾する要件と、シーンの残余部のタイル処理を行う適切な空間との間でバランスが取れるように調整可能である。

【0026】

イメージデータのフレームのタイリングが終了すると、次に、システムは、前のフレームのレンダリングが依然とし進行中であっても、新しいフレームのタイル処理を開始できる。新しいフレームのマクロタイルに関連する割り当てブロックは、前のフレームのマクロタイルに関連する割り当てブロックとは異なっている必要がある。このことは、前のフレームが使用したブロックとは独立した第2のセットのブロックを保持することによって達成できる。このことは、第2のセットの領域ヘッダを独立して保持することによって達成できる。全てのフレームに対して割り当てブロックは同一の空き記憶域から取り込まれるが、このことはディスプレイリストのサイズを2倍にする必要がないことを意味している。このことは、更にメモリ使用量を低減する。

【0027】

従って、z/フレームバッファロード/記憶プロセスは、それぞれ各々のレンダリングの始めと終わりに発生することが分かる。レンダリングされたマクロタイルのタイルに対してのみ発生する。最初のレンダリングの前にロード操作は必要でなく、最後のレンダリングの後に記憶操作は必要でない。使用中の割り当てブロックの数が所定の閾値に達し、またはレンダリングが終了した場合に、有効性及びメモリ使用がマクロタイルのフレームバッファへのレンダリングの結果として生じ、レンダリングされるマクロタイルに関連したメモリブロックが解放されて、別のパラメータを記憶するために利用可能になることが理解できる。

【0028】

記憶されたオブジェクトパラメータは、典型的に三角形の座標であり、レンダ

リングされるオブジェクト、オブジェクト色等である。このようなオブジェクトパラメータとオブジェクトポインタの両者は、各々のマクロタイルの割り当てブロックに記憶されていることが分かる。このことは、必要なくなったパラメータ及びポインタを簡単に破棄することを可能にする。メモリ内のパラメータデータ及びポイントデータの近接性は、十分に小さな割り当てブロック、又は大きなブロックの適切な配置を使用する場合、メモリシステムのページ区切り数の大幅な低減をもたらす。ページ区切りは、メモリシステムの性能を低下させ、メモリハードウェアへの連続的なアクセスが同一ページ内でないときに引き起こされる。可能な限り同一ページにオブジェクトポインタ及びパラメータデータを記憶することは、ページ区切りを最小限にする有効な方法である。

【0029】

割り当てブロックのポインタ及びパラメータのレイアウトから種々の方式が可能である。幾つかの例を、図4a、図4b、及び図4cに示す。最も簡単な方式は、割り当てブロックの先頭の小さなポインタブロックにポインタを割り当て、読み取り／書き込みを最適化する。ブロック内の最後のポインタは、リンクリストを必要とする次のポインタブロックへのリンクである。図4aにおいて、ブロックの一端からポインタが割り当てられ、パラメータは他端から割り当てられている。ブロックが満杯になると両者は真ん中で交わる。図4bにおいて、パラメータとポインタとは交互に配置されており、割り当てブロックは、終始満杯になっている。図4cに示す別の変形例は、1ページより大きな割り当てブロックに対して適切である。オブジェクトポインタが指すパラメータデータと同じデータブロック内にオブジェクトポインタに保持しようとするものであるが、オブジェクトポインタブロックの一部に無駄な空間がある。

【0030】

割り当てブロックはマクロタイルと関連があり、各々のマクロタイルは複数のタイルを含むので、各々の割り当てブロック内には、オブジェクトポインタブロックの複数のリンクリストがある。

【0031】

シーンが大きなオブジェクトから構成され、各々のマクロタイルが多数のタイ

ルを含む場合、割り当てブロックの大きな空間は、大部分は空のポインタブロックにより消費され、そのポインタは全て同じオブジェクトを指していることが分かる。

【0032】

図5は、オブジェクトが見えるマクロタイルのタイルを指示するタイルマスクを備える単一のリストを使用する方式を示す。ストリップマスクは、オブジェクトポインタとは切り離されており、一箇所に集められている。イメージ合成プロセッサのパラメータを取り出すステップは、タイルマスクで設定されたビット数をカウントすることによって関連ストリップマスクへスキップできる。

【0033】

前述の全ての割り当て方式において、ポインタは、常に、同一の割り当てブロック内のパラメータデータを指しているので、適切な最下位ビット数を使用して表すことによって、オブジェクトポインタを圧縮することが可能である。ブロック外（例えば、グローバルリスト内）のオブジェクトのポインタは、同様に供給されるアドレスの最上位ビットをもつ必要がある。これは、大きなポインタを使用することで可能になる。

【0034】

非常に大きな三角形又はストリップ、即ち、多くのマクロタイルを横切るものは前述の方式では問題が起きる場合もある。これは、データが多くのマクロタイルにおいて複製されるので、大量のメモリを使用するからである。この問題の1つの解決策は、大きなオブジェクトを記憶するグローバルリストを生成することである。オブジェクトの順番を確実に記憶するために、これらのオブジェクトとの関連性が、通常の方法で各々のマクロタイルのポインタリストに書き込まれる。

【0035】

オブジェクトは、特定の閾値よりも大きい複数のマクロタイルで見える場合は、大きいとみなすことができる。この閾値は、パラメータの複製、グローバルリストにより使用されるメモリ、及びパラメータメモリの再利用との間の最適なトレードオフを達成するよう選択できる。グローバルリストのパラメータデータは

、多くのマクロタイルのポインタリストで参照することができるので比較的長期間に亘ってメモリ内に残っている。従って、グローバルリストは、マクロタイリング操作に悪影響を与えるほど成長しないようにすることが重要である。大ざっぱな解決策は、グローバルリストのサイズが閾値を超えた場合に、全てのマクロタイルをレンダリングすることである。これにより、グローバルリストに関連する全ての割り当てブロックを空き記憶域に戻すことができる。緻密な方式では、参照カウントを用いることができ、それによって、割り当てブロックを参照するマクロタイルがレンダリングされると即座にグローバルリスト割り当てブロックを解放することができる。

【0036】

次に、どのマクロタイルをレンダリングするかを選択は、一般的に、解放されて空き記憶域に戻すことができる最大量のメモリによって決定される。これにより、マクロタイルが複数のレンダリングの間で適度な複雑性を生み出し得ることを保証することにより効率が高くなり、必要な z /フレームバッファロード/記憶操作の総数が低減する。

【0037】

ディスプレイリストが、複数のフレームのマクロタイルに関連するブロックを含む場合、1つのフレームに属する全てのタイルは、次のフレームの任意のタイルをレンダリングする前に、完全にレンダリングしておくことが重要である。

使用中のマクロタイルがレンダリング中であり、同時に、タイルアクセラレータが同一のマクロタイルに追加のデータを書き込み中である場所は容易に想像できる。これは、画面の小領域が特に詳細なオブジェクトを含む場合に起こり得る。このような状況では、同一のものを繰り返すために、次のマクロタイルを選択してレンダリングを行う選択に重み付けをすることが道理に合っている場合もある。これにより、マクロタイル内の少なくとも1つのタイルに対する、又は小さなオンチップ・キャッシュが存在する場合は2つ以上に対する z /フレームバッファロード/記憶を省略できる。最大の利点を得るには、任意の順番で、マクロタイルのタイルをレンダリング可能になっている必要がある。

【0038】

グローバルリストが厄介なほど大きくなる場合、前述の参照カウント方式を拡張すると、できるだけ多数のグローバルリスト割り当てブロックを解放するレンダリングを得るためのマクロタイルを選択可能にできる。

【0039】

別の実施方法は、漸進的にサイズが大きくなるネスト化マクロタイルを使用して、階層的マクロタイルシステムを使用することである。この目的は、オブジェクトが現れるタイル数に応じて、記憶装置に対する適切な階層レベルを選択することによって、パラメータデータの複製を低減することである。オブジェクトパラメータは、オブジェクトの順番を記憶するためのみに、最低限のレベルで記憶され得る。

【0040】

前述のマクロタイリングによるメモリ管理手法は、システムの帯域幅要件をかなり低減するが、 z / フレームバッファロード / 記憶を完全になくすことはできない。また、記憶された z 情報は、追加的にメモリ空間を占有することに留意されたい。 z 情報及びフレームバッファデータの圧縮は、メモリ帯域幅及び記憶要件を更に低減する。

【0041】

種々の圧縮手法を利用でき、記憶データによって占有される空間を低減する。種々の圧縮方法の効率は、圧縮データの性質に影響を受けるので、単一の方式が、全ての状況において有効な圧縮をもたらす見込みはない。従って、階層的システムが提案されており、システムは、その時点での方法が、その時点でのデータセットに適切でない場合は、別の圧縮方法に移る。もしくは、場合によっては不十分なデータ圧縮（又は展開さえも）を与えることは承知の上で、提案方法のいずれか1つを分離して使用することができる。

【0042】

1) z 圧縮のための面の平面方程式の記憶

タイルの z 値は、各々の画素において見える三角形の奥行きを含む。各々のタイルに対して処理される三角形は数多く存在する場合もあるが、殆どの場合は少数の三角形のみが見えるので、奥行き値をもたらす。数多くのイメージの最初の

部分に発生する z ークリアポリゴンの場合、1つのポリゴンは、タイル全体の初期の z 状態を定義することができる。

【0043】

所定のポリゴンについて、 z 情報は、3つの平面パラメータA、B、及びCによって定義され、ISPによって計算される。パラメータは、式、即ち、奥行き $=Ax + By + C$ に使用される。ここで、 x 、 y は該当の画素位置であり、(0、0)はタイルの左上コーナにある。

【0044】

パラメータA、B、及びCの記憶は、三角形毎に12バイトのデータが必要である。追加のビットマップ値は画素毎に記憶され、異なるセットのパラメータの間で選択するようになっている。 z 奥行きを4つのポリゴンによって定義した例を図6に示す。

【0045】

データの圧縮には、幾つのポリゴンが z 値を与えるかを知ること、及びポリゴンの平面パラメータを呼び出す能力が必要である。このデータは、データがTSPに出力される点での記憶されるオブジェクト識別タグから計算できる。9個以上のポリゴンが奥行きを与える場合、システムは、以下に説明する別の方式の1つに移ることができる。

【0046】

32 x 16画素の矩形タイルの奥行き情報を与える種々の数の三角形に関する圧縮率は、以下の通りである。

ポリゴン数	ISPデータサイズ	ビットマップデータサイズ	総データサイズ
1	12バイト	0	12バイト
2	24バイト	64バイト	88バイト
3から4	48バイト	128バイト	176バイト
5から8	96バイト	256バイト	352バイト

【0047】

浮動小数点 z の生データは、2048バイトを必要とする。従って、この方法では、5倍から128倍のかなりの圧縮率が得られることが分かる。

【0048】

これらの奥行きデータを再ロードするには、三角形を処理する際にISPが行なう計算と類似の計算を必要とする。このようにして計算した奥行き値は、元のセットの値を再構成するためにビットマップデータに基づいて奥行きバッファに書き込まれる。

【0049】

前記の例は、 z 奥行きを、1から8までの三角形によって定義し得るが、他の範囲を使用することもできる。三角形の数がこの範囲外にある場合、システムは、以下で説明する別の方式に移ることができる。

【0050】

2) z 圧縮のランレングス符号化

ランレングス符号化は、一般的に、データが同一の値の多数の連続的な項目を含む場合に使用される。数字は対になって記憶され、その結果、例えば、符号化8:0.5、24:0.8では、第1の8は0.5の奥行き値をもち、次の24は、異なる三角形に属すると想定して、0.8の奥行き値をもつ画素の列を示す。

【0051】

別の種類のランレングス符号化は、通常、奥行き情報を符号化するのに有効なことが判明する場合もあり、三角形は、「平面」ではないので、タイルに沿って均一に変化する奥行き値をもつ可能性がある。この状態を図7に示す。三角形は平面的なので、「デルタ」値（1つの画素と次の画素との奥行きの差）が一定であることがわかる。実際には、タイルに沿って x 方向に移動すると、デルタ値は、そのポリゴンに関するISPの「A」パラメータに等しい。タイル線に沿って x デルタ値をランレングス符号化する（定数 y ）と、一般的なデータに対して約2倍の圧縮率が得られる（実験結果から）。

【0052】

また、(デルタ値以外の)ランレングス符号化は、ステンシル及び「タグ」(特定の画素位置におけるオブジェクトの識別)値に非常に適しており、ISPの状態を完全に記録するために、奥行き情報と共に記憶する必要がある。

【0053】

ランレングス符号化の欠点は、データサイズが実際には拡張する状況になる可能性をもつ点である。このことは、例えば、何れの隣接画素も同一でない場合に起こり得る。各々の値が1度しか発生していないことを示すカウンタを記憶する必要があるので、圧縮されたデータは、実際にはサイズが2倍になる場合もある。この場合、z圧縮システムは、圧縮されていないz値を記憶する方法に移ることになる。

【0054】

3) フレームバッファ圧縮

フレームバッファメモリは、ビデオイメージの生成ハードウェアによって直接読み取られるので、最終出力は圧縮できない。しかし、データが、イメージを組み立てるプロセスにおいて複数回記憶されてロードされる場合、中間データを圧縮することによってメモリ帯域幅を記憶できる。ランレングス符号化、J-P E G、W a v e l e t、ベクトル定量化(VQ)を含む、標準的なイメージ圧縮手法を使用できる。

【0055】

4) 変化時にのみZを書き込み、要求時にのみTSPを読み取る

最後の最適化は、特定のz記憶及びフレームバッファロードを回避することを可能にし得る。

「ダーティー」フラグは、データが最後に読み取られてから変更されたことを示す。zバッファロード/記憶の場合、zロードが起こるとダーティーフラグはリセット状態になり、zデータがISPにより変更されるとセットされる。z記憶の要求があると、ダーティービットがセットされた場合に実際にデータを記憶することのみが必要である。

【0056】

類似の最適化は、フレームバッファロード/記憶に適用できる。特定の混合操

作に必要な場合に、フレームバッファから画素をロードすることのみが必要であり、この状況は常に起こるわけではない。同様に、ダーティーフラグは、変更された画素又は画素群を示し、それらの画素のみがフレームバッファ記憶操作により転送される。

【0057】

5) アプリケーションが必要な場合に「z バッファ」にアクセス可能にする

Imagination Technologies Power Vr System等の特定のレンダリングシステムは、通常、全フレームに対するz バッファを保持する必要はない。これはメモリ及び帯域幅を記憶する。しかし、特定のアプリケーションは、z バッファを見ようとし、特別のグラフィック効果を得るために自ら操作しようとする。z バッファ記憶操作により記憶されるデータは、従来の意味ではz バッファではないが、メモリ内のデータのレイアウトを処理することによってアプリケーションには、そのようなものとして見えるようにすることが可能である。必要であれば、記憶されたz 値が適切な場合に仮想のz バッファの状態を確実に反映するように、追加のz バッファ記憶操作を行うことができる。

【0058】

従来のz バッファは、圧縮されていない生のz データを含む。前述の方法を利用して圧縮z バッファを読み取ろうとするアプリケーションには、正確に解釈できないことは明らかである。従って、z バッファを必要とするアプリケーションに対して圧縮を解除するか、又は「仮想z バッファ」をアプリケーションに与えて透過的に圧縮及び圧縮解除を行う追加のハードウェアを提供する機能があることが重要である。

【0059】

本発明の実施形態を示す図を以下に説明する。図8は、本発明が適用されるシステム形式のブロック図を示す。これは、図1の僅かに拡張されたバージョンを包含し、ディスプレイリストメモリ4に供給されるタイリング操作を行うタイルアクセラレータ2を示している。次に、これは、光線／面交差テストを行うイメージ合成プロセッサ6にデータを供給する。次に、このデータは、フレームバッ

ファメモリ11にテクスチャリングされたイメージデータを記憶する前に、メモリ9のテクスチャからのテクスチャリング及びシェイディングデータを用いるテクスチャリング／シェイディングプロセッサ8に供給される。この装置は、図1を参照して行った前述の説明に基づいて作動する。

【0060】

図9は、図8のz／フレームバッファロード／記憶操作を行う回路が変更されたシステムを示す。このシステムでは、z／フレームメモリ16は、z圧縮／圧縮解除ユニット18を経由してイメージ合成プロセッサ6と接続している。これは、システムが複雑なシーンをレンダリング中であり、ディスプレイリストメモリが、特定のタイルに対して処理される必要がある全ての面を含むほど大きくない場合に作動する。このプロセスは前述の通りである。しかし、図9のシステムを使用して作動する場合、ディスプレイリストは、タイルアクセラレータによって、実質的に満杯になるまで全てのタイルに関するデータがロードされる。しかし、これは、初期データの一部を表しているに過ぎない場合もある。このイメージは、イメージ合成プロセッサ6によって、一度に1つのタイルがレンダリングされる。各々のタイルの出力データは、テクスチャメモリ9からのデータを使用してタイルをテクスチャリングし、フレームバッファメモリ11へ供給する、テクスチャリング／シェイディングプロセッサ8に供給される。

【0061】

同時に、イメージデータが不完全であった場合は、イメージ合成プロセッサ6からの結果が圧縮／圧縮解除手段18を経由してz／フレームメモリ16に一時的に記憶される。全てのタイルがレンダリングされてフレームバッファメモリ及びz／フレームメモリ16に記憶されるまで、残りのタイルのレンダリングが不完全なイメージデータの状態で継続される。

次に、ディスプレイリストの第1の部分が破棄されて、追加のイメージデータがディスプレイリストに読み込まれる。イメージ合成プロセッサ6により順番に各々のタイルに対して処理が行なわれる場合、ディスプレイリストメモリ4の新しいイメージデータと組み合わせることができるよう、z／フレームメモリ16のデータの関連部分がz圧縮／圧縮解除手段18を経由してロードされる。次

に、各々のタイルの新しいデータが順番に、フレームバッファメモリ11に供給される前に、テクスチャメモリ9のデータと組み合わせるテクスチャリング／シェイディングプロセッサ8に供給される。

【0062】

このプロセスは、シーン内の全てのタイルに対して、全てのイメージデータがレンダリングされるまで継続する。

【0063】

従って、zバッファメモリは一時的記憶装置を満杯にして、特に複雑なシーンをレンダリングするのに必要なものよりも小さなディスプレイリストメモリを使用可能にすることが分かる。圧縮／圧縮解除手段18は随意的であるが、それを使用すると更に小さなzバッファメモリを用いることができる。

【0064】

テクスチャリング／シェイディングプロセッサとフレームバッファメモリとの間の追加リンクとして、フレームバッファ圧縮／圧縮解除手段20が設けられている。これは、フレームバッファメモリ内の不完全なイメージデータの一時的な記憶と、既にテクスチャリング／シェイディング処理されたデータと組み合わせる必要があるタイルに関して、イメージ合成プロセッサ6から追加データが供給された際に、そのデータをフィードバックするのに使用するのが好ましい。勿論、圧縮／圧縮解除手段は随意的なものである。

【0065】

図10において、タイルアクセラレータによって使用されるメモリ管理システムを示す。これは図3の説明を参照して理解されたい。ステップ30では、領域ヘッダを初期化し、ステップ32では、システムは、領域ヘッダにオブジェクトデータが入力されるのを待つ。ステップ34では、タイル及びマクロタイルが図3に示すように割り当てられ、ステップ36では、新しいメモリブロックが必要か否かを判定する。追加のメモリブロックが必要ない場合、マクロタイルのオブジェクトデータが、システムにより先に割り当てられたメモリブロックに記憶されるが、ステップ40では、マクロタイル及びタイルに対するオブジェクトデータの割り当て領域及びメモリにおける、フレーム内の最後のオブジェクトに達し

たか否かを決定する。ステップ42では、フレームに関してレンダリングされていないマクロタイルがあるか否かを判定する。レンダリングされていないマクロタイルがある場合、ステップ44でマクロタイルを選択してレンダリングする。ステップ48で追加オブジェクトを記憶してレンダリングする必要があるか否かを判定する前に、ステップ46でレンダリングを開始する。ステップ48で「N O」の場合、ステップ42で、別のタイルをレンダリングすべきか否かを判定し、更なるマクロタイルがない場合、ステップ50で、別のフレーム又はデータを入力すべきか否かを判定する。別のフレーム又はデータを入力する必要がある場合、システムは、フローチャートの最初に戻る、別のフレーム又はデータを入力する必要がない場合、本プロセスは終了する。ステップ42での判定結果が「Y E S」の場合、制御は、メモリからのロードすべき新しいブロックに関する図6に進む。

【0066】

ステップ36で追加メモリが必要であると判定された場合、ステップ52でメモリ満杯閾値に既に達しているか否かを判定する。メモリ満杯閾値に達していない場合、ステップ54で新しいメモリブロックを空き記憶域から要求し、ステップ38で、前述のようにステップ40へ進む前に、マクロタイルのオブジェクトデータを記憶するのに使用される。メモリ満杯閾値を超えている場合、プロセスは、マクロタイルをレンダリングするステップ44に進む。このルートがステップ48に進むと、システムは、オブジェクトを記憶するのを待つか否かを判定する。この場合、ステップ38でマクロタイルの追加オブジェクトを記憶するために、ステップ54で要求を行う前に、メモリブロックが利用可能になるまで待つステップ56に進む。

【0067】

図11は、タイルアクセラレータによって、どのメモリブロックが解放されてその後の割り当てのために記憶装置に戻されるかを示す。ステップ80で、システムは起動して、部分レンダリングが終了した信号を待つ。この信号を受け取ると、ステップ82でメモリブロックは記憶装置に戻され、システムは、次の部分レンダリングを待つ。

【0068】

タイリングとレンダリングとは並行して行われるので、図10ではメモリブロックの解放は個別に示されており、ステップ46では、信号をISPに送ることによってレンダリングが始まる。タイルアクセラレータはその時点で多くのタイリング操作を行い得るので、レンダリングの終了を待たない。これがマクロタイルのレンダリングが開始される閾値（例えば79%）が存在する理由である。これにより、タイルアクセラレータが作業を行うメモリ空き領域が残る。

【0069】

図11のシステムは、タイルアクセラレータがサイクル中のどのステップにあるのか分からず、それに割り込みたくないので、ISPが、部分レンダリングの終了を知らせるのを待つ。図11は、図10のステップ56及び54に達した場合に可能な全てのメモリを残す。そこにはメモリが使用に向けて直ちに利用可能になるよいチャンスがある。メモリは、常に利用可能とは限らないので時としてステップ56で待つ必要もある。このことは複雑なイメージの場合に起こる場合もあり、メモリが利用可能になるのを待つ必要がある。

【0070】

図12は、図8及び図9のISP8及びディスプレイリストメモリ4を示すが、タイルアクセラレータ2は更に詳細に示されている。入力データ90は、タイリングユニット92に送られ、次に、このデータは、マクロタイリングユニット94に送られる。出力データ95は、ディスプレイリストメモリ4及びディスプレイリストメモリマネージャ96へ送られる。メモリマネージャ96は、空き記憶域リスト98及びディスプレイリスト4と通信し合う。例えば、メモリが75%使用中であることを示す信号をディスプレイリスト45から受信し、これに応答してレンダリング開始信号をISP6へ送る。これが終了すると、レンダリング終了信号は、メモリマネージャ96へ送られ、更なるマクロタイルによって使用できる現在利用可能なメモリを空き記憶域リスト98に戻す。メモリマネージャは、先のマクロタイルがレンダリングされている間に、マクロタイルデータの受信を続け、空き記憶域リストのメモリを、必要に応じてオブジェクトに割り当てる。

【図面の簡単な説明】**【図 1】**

レンダリング及びテクスチャリングシステムのブロック図である。

【図 2】

公知のタイリングシステムにおいて、ディスプレイリストが如何に順序付けされているか概略的に示す。

【図 3】

本発明のマクロタイリングにより、ディスプレイリストが如何に順序付けされているか概略的に示す。

【図 4 a】

本発明の実施形態による、ディスプレイリストのデータブロックの割り当てに関する別の配置を示す。

【図 4 b】

本発明の実施形態による、ディスプレイリストのデータブロックの割り当てに関する別の配置を示す。

【図 4 c】

本発明の実施形態による、ディスプレイリストのデータブロックの割り当てに関する別の配置を示す。

【図 5】

ディスプレイリストの別の実施形態を示す。

【図 6】

4つのポリゴンの z 奥行きと、4つのポリゴンの平面パラメータの記憶による z 圧縮を概略的に示す。

【図 7】

ランレングス符号化の結果を示すグラフである。

【図 8】

本発明で利用できるレンダリングシステム形式の例についてのブロック図である。

【図 9】

本発明の 1 つの態様の実施形態における図 8 のシステムを示す。

【図 10】

本発明の実施形態で使用するメモリ管理を示すフローチャートである。

【図 11】

図 10 のフローチャートのサブプロセスを示すフローチャートである。

【図 12】

本発明の実施形態で使用するタイルアクセラレータのブロック図である。

【図1】

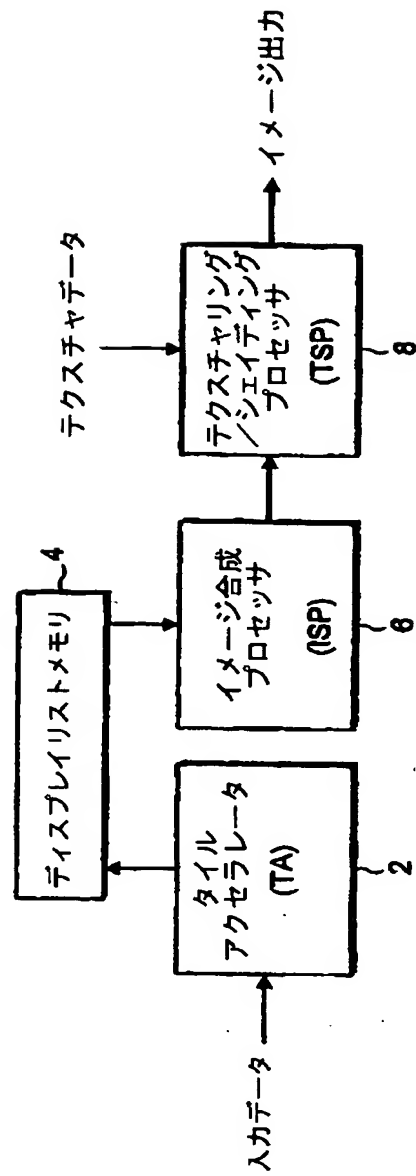


FIG. 1

【図2】

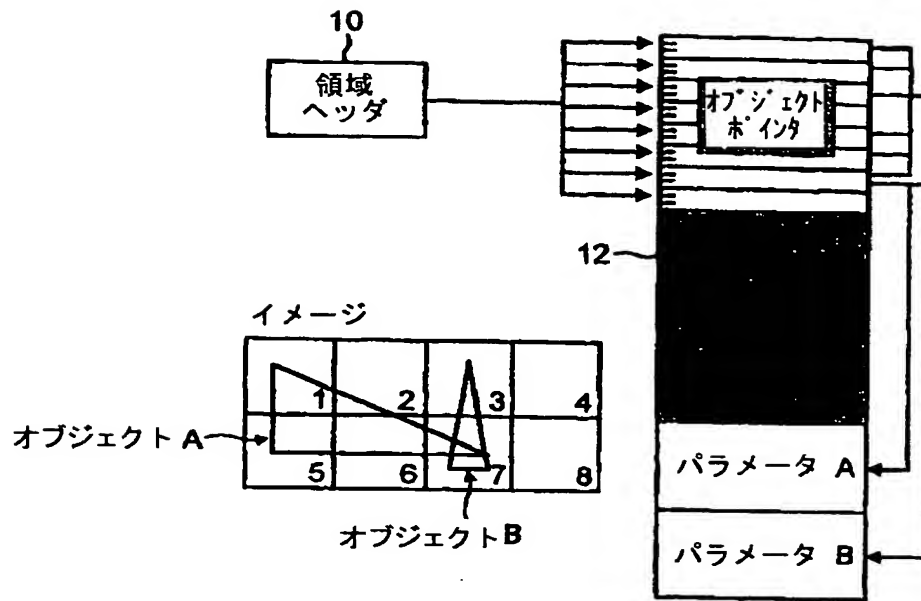


FIG. 2

【図3】

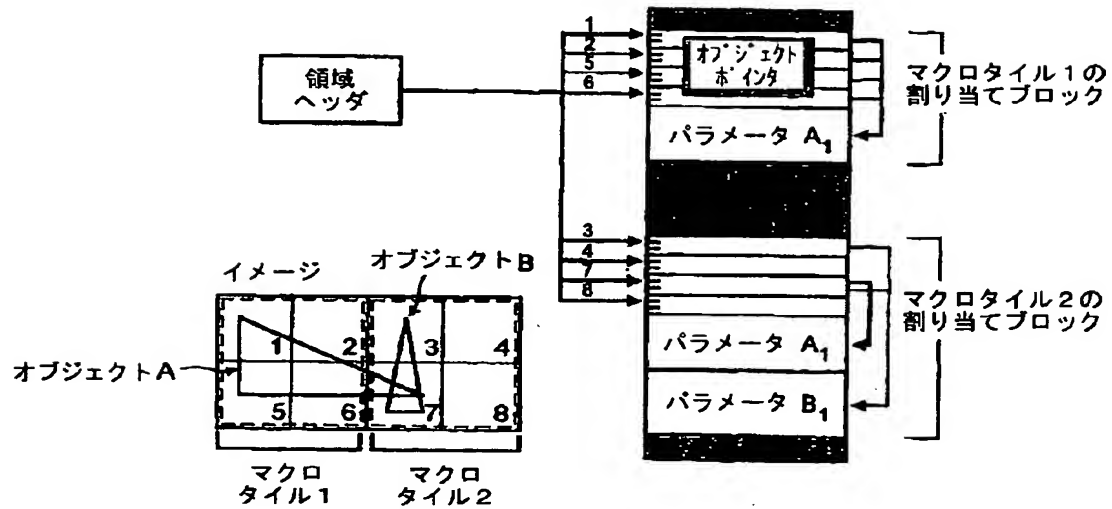


FIG. 3

【図4A】

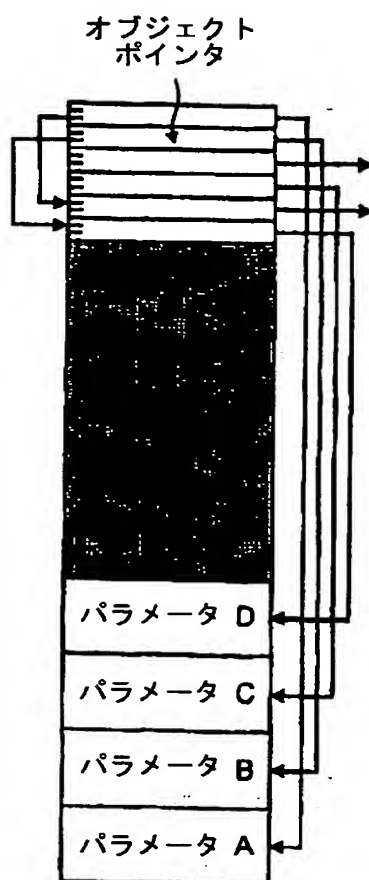


FIG. 4A

【図4B】

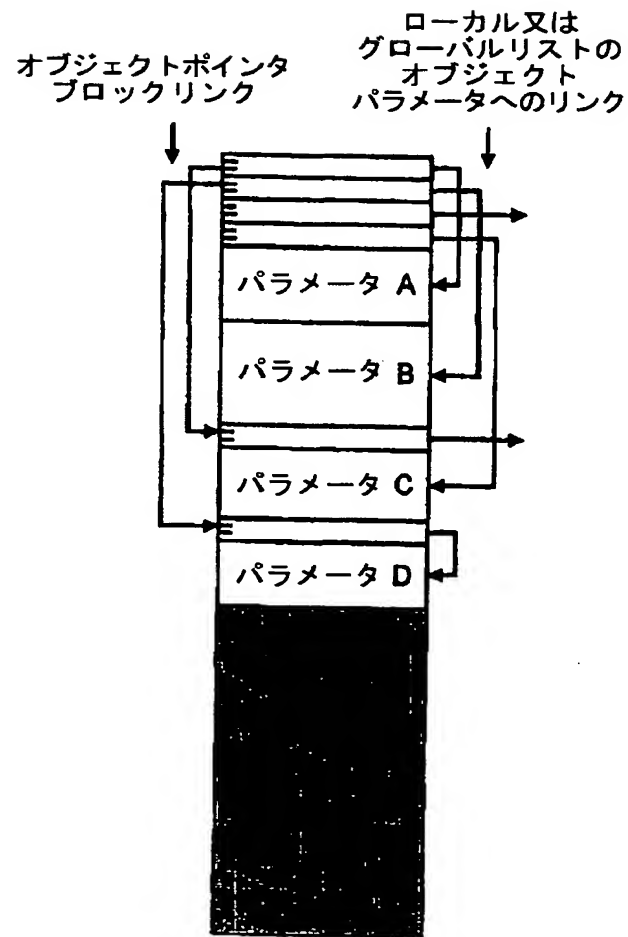


FIG. 4B

【図4C】

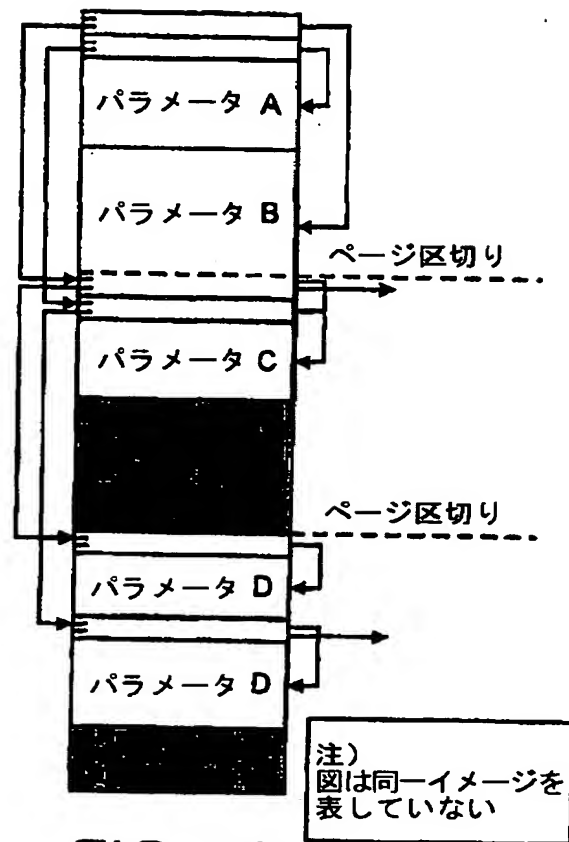


FIG. 4C

【図5】

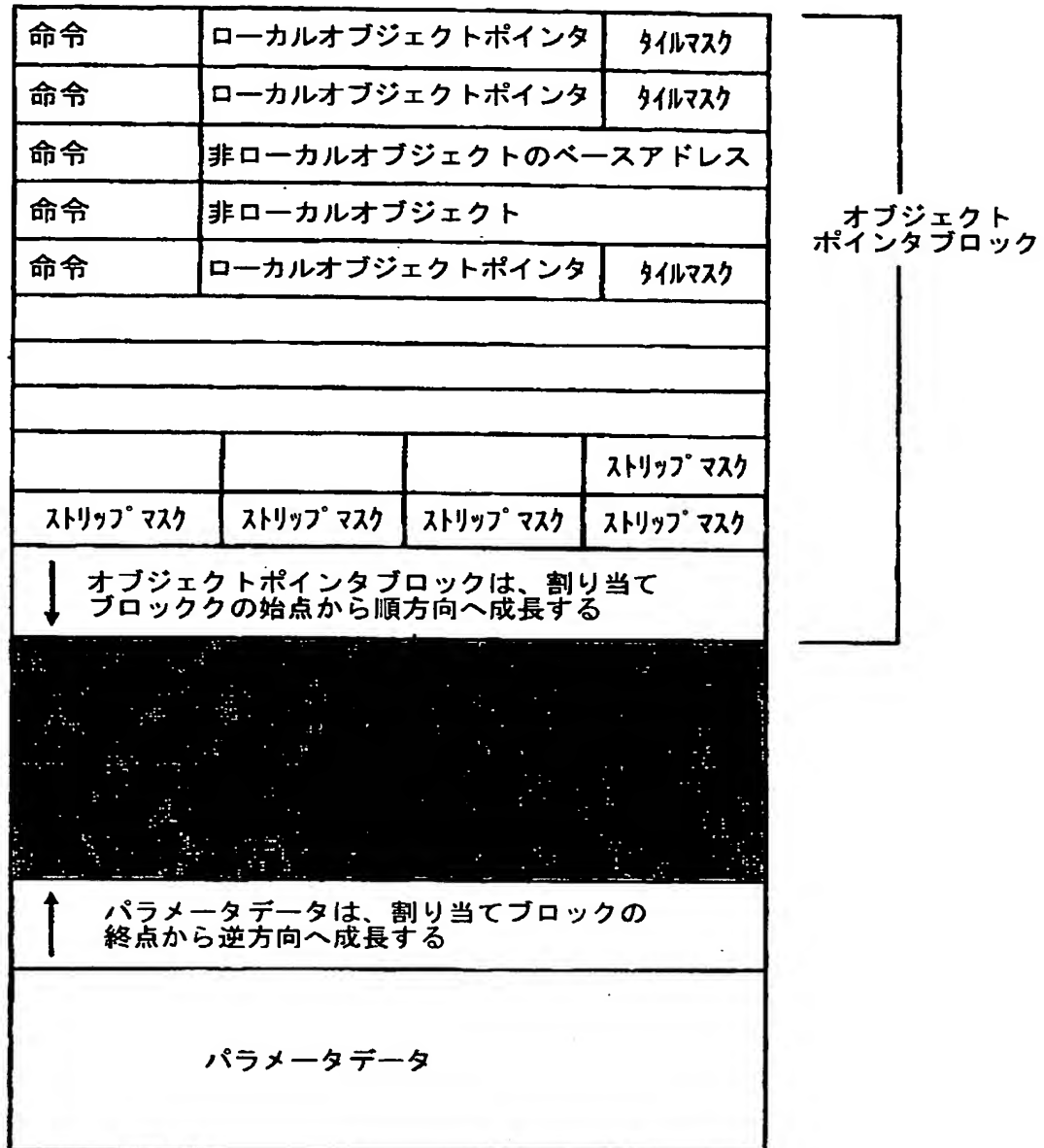


FIG. 5

【図6】

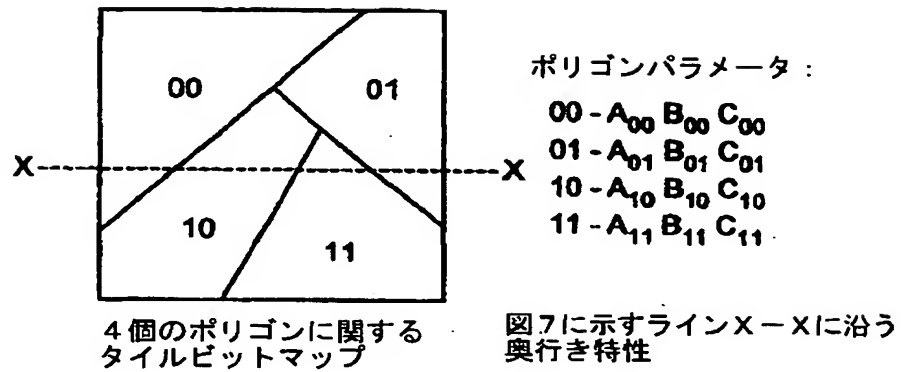


FIG. 6

【図7】

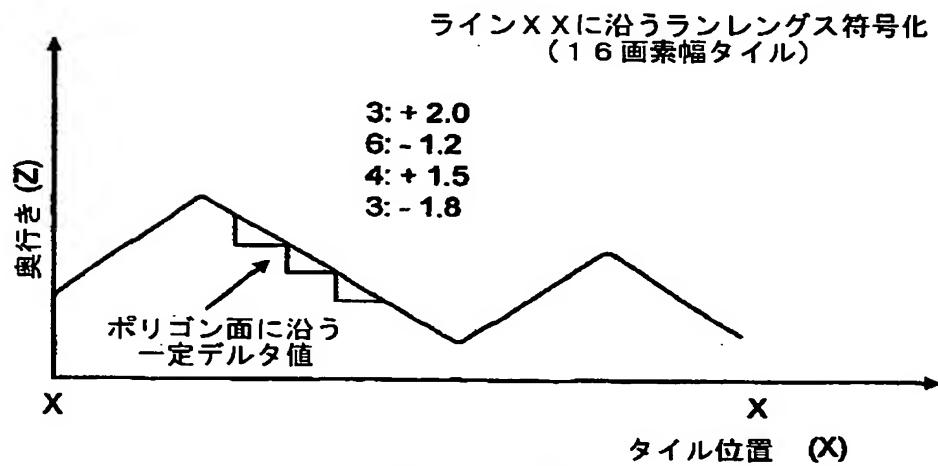


FIG. 7

【図8】

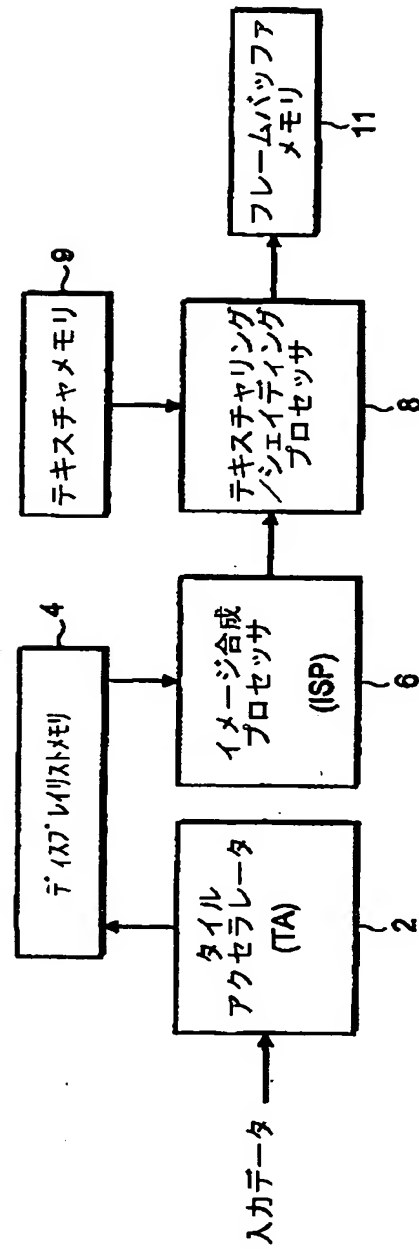


FIG. 8

【図9】

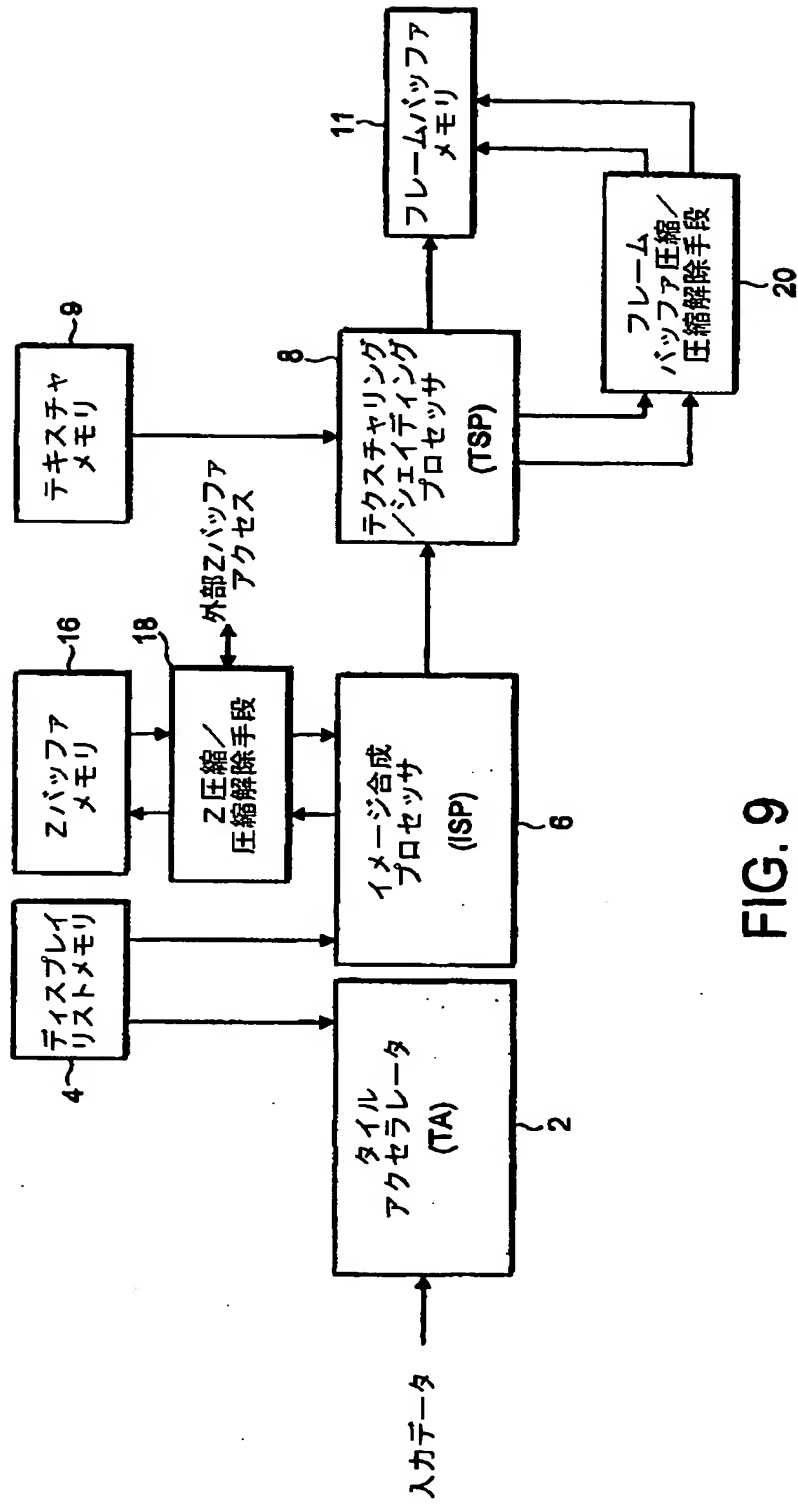


FIG. 9

【図10】

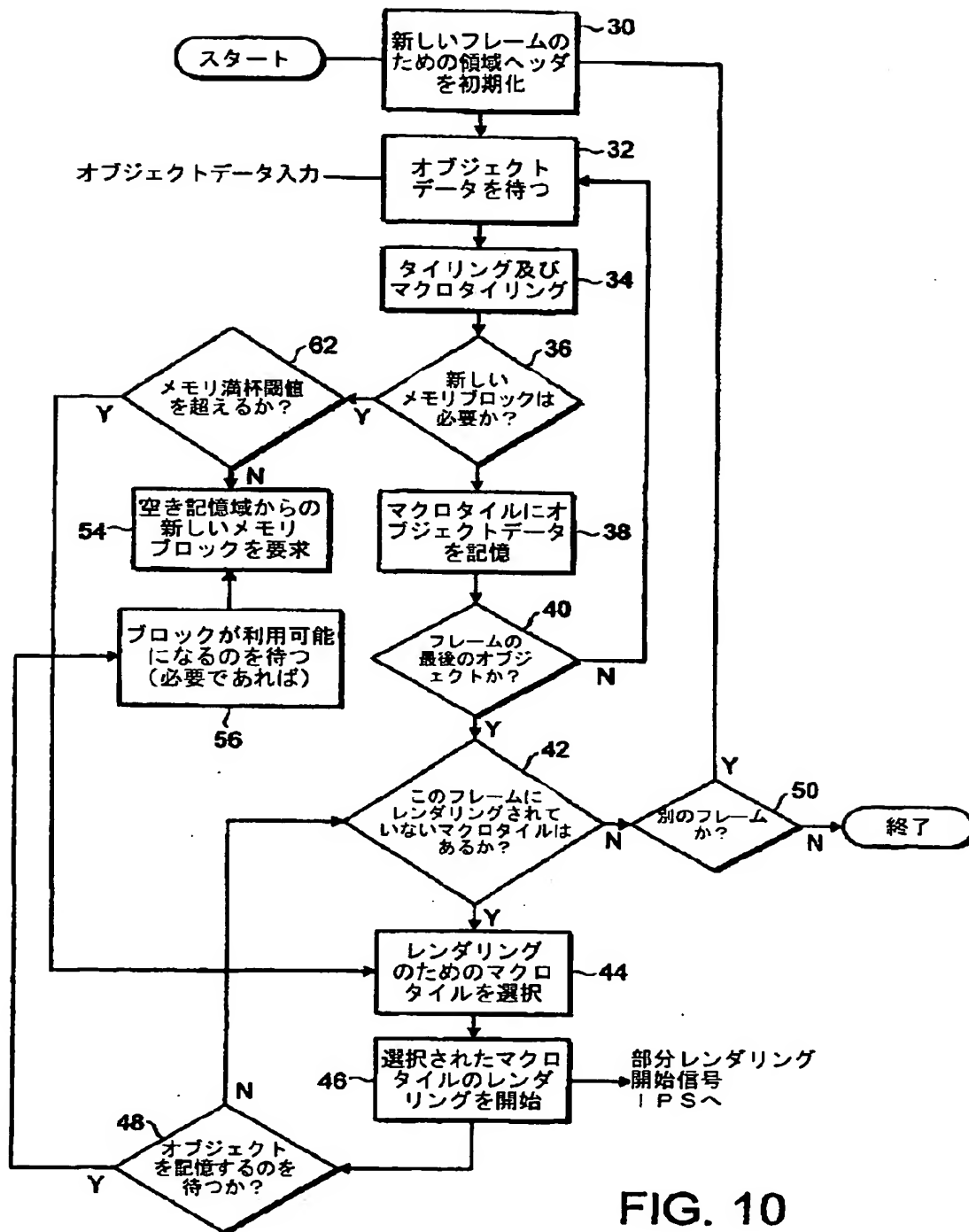


FIG. 10

【図11】

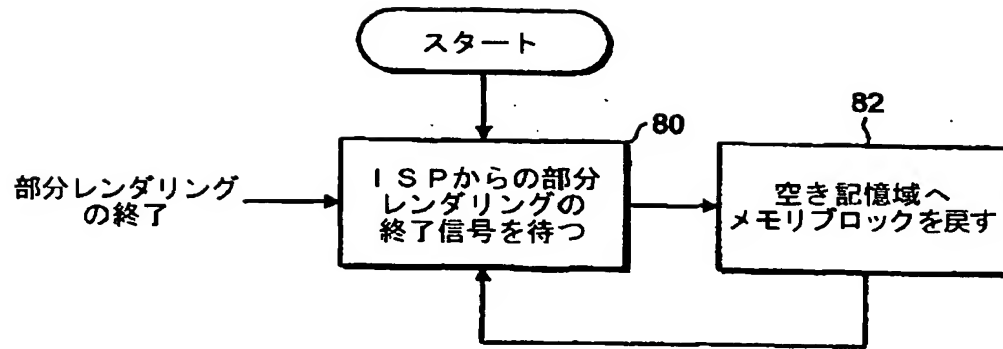


FIG. 11

【図12】

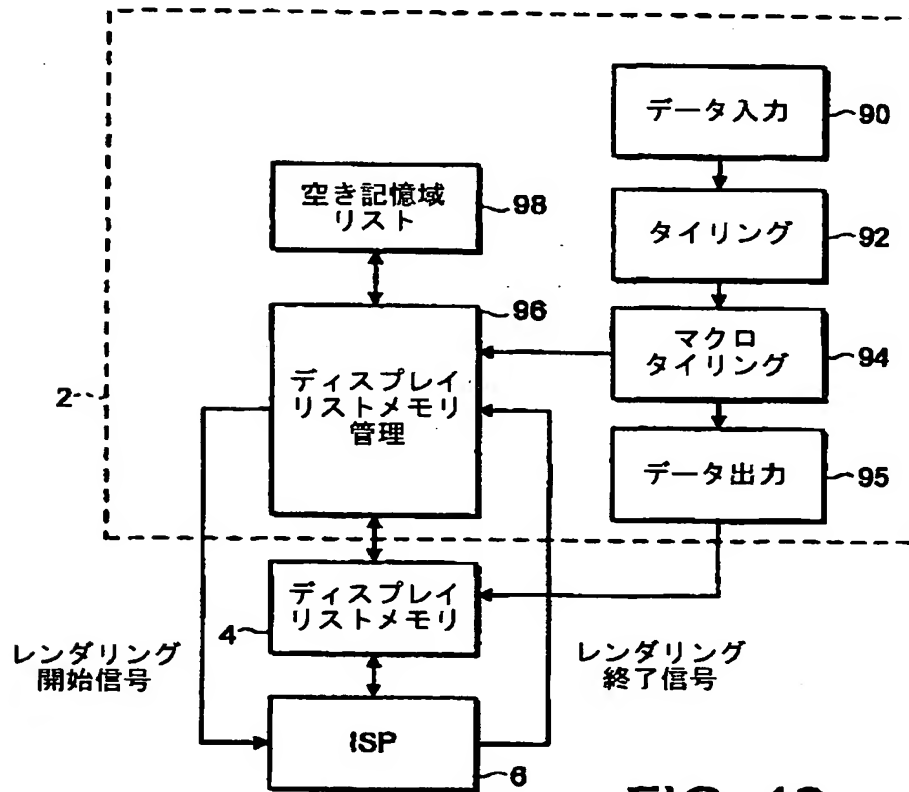


FIG. 12

【国際調査報告】

INTERNATIONAL SEARCH REPORT

 Intern Application No
 PCT/GB 01/02536

A. CLASSIFICATION OF SUBJECT MATTER IPC 7 G06T15/00		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) IPC 7 G06T		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, WPI Data, INSPEC, PAJ		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 886 701 A (POWELL III WILLIAM CHAMBERS ET AL) 23 March 1999 (1999-03-23) abstract; figures 9,20 column 5, line 25 -column 6, line 16 column 8, line 11 - line 62 column 9, line 36 -column 10, line 37 column 16, line 13 - line 60 column 61, line 61 -column 63; line 20	1-13
A	US 5 392 385 A (EVANGELISTI CARLO J ET AL) 21 February 1995 (1995-02-21) figures 2,7 column 3, line 7 -column 5, line 19	1
A	US 5 522 018 A (TAKEDA MASAKI ET AL) 28 May 1996 (1996-05-28) figure 9	2
-/-		
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C.		
<input checked="" type="checkbox"/> Patent family members are listed in annex.		
* Special categories of cited documents:		
A document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) *O* document relating to an oral disclosure, use, exhibition or other means *P* document published prior to the international filing date but later than the priority date claimed *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art *Z* document member of the same patent family		
Date of the actual completion of the international search 5 September 2001		Date of mailing of the international search report 12/09/2001
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patenkamp 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3018		Authorized officer Diallo, B

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inventor Application No
PCT/GB 01/02536

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 295 235 A (NEWMAN STEVE) 15 March 1994 (1994-03-15) abstract; claims 1,2,4 column 2, line 26 - line 60 column 6, line 54 -column 7, line 28	3,6
A	US 5 729 672 A (ASHTON MARTIN) 17 March 1998 (1998-03-17)	

INTERNATIONAL SEARCH REPORT

Information on patent family members

Intern. Application No.
PCT/GB 01/02536

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5886701 A	23-03-1999	AU 6766096 A	05-03-1997
		CA 2229027 A	20-02-1997
		EP 0850462 A	01-07-1998
		JP 11511277 T	28-09-1999
		WO 9706512 A	20-02-1997
		US 6252608 B	26-06-2001
		US 5977977 A	02-11-1999
		US 5880737 A	09-03-1999
		US 5808617 A	15-09-1998
		US 5867166 A	02-02-1999
		US 5864342 A	26-01-1999
		US 5852443 A	22-12-1998
		US 5870097 A	09-02-1999
		US 5999189 A	07-12-1999
		US 6005582 A	21-12-1999
		US 5990904 A	23-11-1999
		US 5949428 A	07-09-1999
		US 6008820 A	28-12-1999
		US 6064393 A	16-05-2000
		US 6016150 A	18-01-2000
US 5392385 A	21-02-1995	CA 1309198 A	20-10-1992
		DE 3855377 D	25-07-1996
		EP 0319787 A	14-06-1989
		JP 1191985 A	02-08-1989
US 5522018 A	28-05-1996	JP 2807608 B	08-10-1998
		JP 6203172 A	22-07-1994
US 5295235 A	15-03-1994	NONE	
US 5729672 A	17-03-1998	GB 2298111 A	21-08-1996
		GB 2281682 A, B	08-03-1995
		JP 7152926 A	16-06-1995
		US 5596685 A	21-01-1997
		DE 69609534 D	07-09-2000
		DE 69609534 T	07-12-2000
		EP 0725367 A	07-08-1996
		ES 2150071 T	16-11-2000
		GB 2336982 A, B	03-11-1999
		GB 2336983 A, B	03-11-1999
		JP 8255262 A	01-10-1996

フロントページの続き

(81)指定国 EP(AT, BE, CH, CY,
DE, DK, ES, FI, FR, GB, GR, IE, I
T, LU, MC, NL, PT, SE, TR), GB, J
P

(72)発明者 モーフェット スティーブン
イギリス ハーツ エイチピー3 9エス
エイチ ヘメル ヘンプステッド ウェイ
マス ストリート 80エイ

Fターム(参考) 5B080 AA14 CA01 CA08 GA02 GA11
GA22

【要約の続き】

のために供給される。少なくとも1つのブロックをディスプレイリストメモリから各々の矩形領域に割り当てた後に、その矩形領域と交差する面に関連するデータをそのメモリブロックに記憶することによりメモリを管理できる。矩形領域について所定数のブロックが使用されている場合を判定する。この数に達すると、システムは、矩形領域のシェイディングデータを引き出すことを開始し、これにより、ディスプレイリストメモリにおいて別の矩形領域に割り当てられる記憶ブロックが解放される。

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.